Is Over Practice Necessary? – Improving Learning Efficiency with the Cognitive Tutor through Educational Data Mining

Hao CEN, Kenneth KOEDINGER, Brian JUNKER
Carnegie Mellon University
5000 Forbes, Pittsburgh, PA, U.S.A.
hcen@andrew.cmu.edu, koedinger@cmu.edu, brian@stat.cmu.edu

Abstract. This study examined the effectiveness of an educational data mining method – Learning Factors Analysis (LFA) – on improving the learning efficiency in the Cognitive Tutor curriculum. LFA uses a statistical model to predict how students perform in each practice of a knowledge component (KC), and identifies over-practiced or under-practiced KCs. By using the LFA findings on the Cognitive Tutor geometry curriculum, we optimized the curriculum with the goal of improving student learning efficiency. With a control group design, we analyzed the learning performance and the learning time of high school students participating in the Optimized Cognitive Tutor geometry curriculum. Results were compared to students participating in the traditional Cognitive Tutor geometry curriculum. Analyses indicated that students in the optimized condition saved a significant amount of time in the optimized curriculum units, compared with the time spent by the control group. There was no significant difference in the learning performance of the two groups in either an immediate post test or a two-week-later retention test. Findings support the use of this data mining technique to improve learning efficiency with other computer-tutor-based curricula.

Keywords: Data mining, intelligent tutoring systems, learning efficiency

Introduction

Much intelligent tutoring system (ITS) research has been focused on designing new features to improve learning gains measured by the difference between pre and post test scores. However, learning time is another principal measure in the summative evaluation of an ITS. Intelligent tutors contribute more to education when they accelerate learning [9]. Bloom's "Two Sigma" effect of a model human tutor [4] has been one of the ultimate goals for most intelligent tutors to achieve. So should be the "Accelerated Learning" effect shown by SHERLOCK's offering four-year's trouble shooting experience in the space of seven days of practice [12].

Cognitive Tutors are an ITS based on cognitive psychology results [11]. Students spend about 40% of their class time using the software. The software is built on cognitive models, which represent the knowledge a student might possess about a given subject. The software assesses students' knowledge step by step and presents curricula tailored to individual skill levels [11]. According to Carnegie Learning Inc., by 2006, Cognitive Tutors have been widely used in over 1300 school districts in the U.S. by over 475,000 secondary school students. With such a large user base, the learning efficiency with the Tutor is of great importance. If every student saves four hours of learning over one year, nearly two million hours will be saved. To ensure adequate yearly progress, many schools are calling for an increase in instructional time. However, the reality is that students have a limited amount of total learning time, and teachers have limited amount of instructional time. Saving one hour of learning time can be better than increasing one hour of instructional time because it does not increase students' or teachers' work load. Moreover, if these saved hours are devoted to other time-consuming subjects, they can improve the learning gains in those subjects.

Educational data mining is an emerging area, which provides many potential insights that may improve education theory and learning outcomes. Much educational data mining to date has stopped at the point of yielding new insights, but has not yet come full circle to show how such insights can yield a better intelligent tutoring system (ITS) that can improve student learning [2, 3].

Learning Factors Analysis (LFA) [6, 5] is a data-mining method for evaluating cognitive models and analyzing student-tutor log data. Combining a statistical model [10], human expertise and a combinatorial search, LFA is able to measure the difficulty and the learning rates of knowledge components (KC), predict student performance in each KC practice, identify over-practiced or under-practiced KCs, and discover "hidden" KCs interpretable to humans. The statistical model is shown in Eq. (1).

$$\log\left(\frac{P_{ijt}}{1-P_{ijt}}\right) = \sum \theta_i X_i + \sum \beta_j Y_j + \sum \gamma_j Y_j T_{jt} \tag{1}$$

P_{ijt} is the probability of getting a step in a tutoring question right by the i^{th} student's t^{th} opportunity to practice the j^{th} KC. The model says that the log odds of P_{ijt} is proportional to the overall "smarts" of that student (θ_i) plus the "easiness" of that KC (β_j) plus the amount gained (γ_j) for each practice opportunity. With this model, we can show the learning growth of students at any current or past moment.

By applying LFA to the student log data from the Area unit of the 1997 Geometry Cognitive Tutor, we found two interesting phenomena. On the one hand, some easy (i.e. high β_j) KCs with low learning rates (i.e. low γ_j) are practiced many times. Few improvements can be made in the later stages of those practices. KC rectangle-area is an example. This KC characterizes the skill of finding the area of a rectangle, given the base and height. As shown in Figure 1, students have an initial error rate around 12%. After 18 times of practice, the error rate reduces to only 8%. The average number of practices per student is 10. Many practices spent on an easy skill are not a good use of student time. Reducing the amount of practice for this skill may save student time without compromising their performance. Other over-practiced KCs include square-area, and parallelogram-area. On the other hand, some difficult (i.e. low β_j) KCs with high learning rates (i.e. high γ_j) do not receive enough practice. Trapezoid-area is such an example in the unit. But students received up to a maximum of 6 practices. Its initial error rate is 76%. By the end of the 6th practice the error rate remains as high as 40%, far from the level of mastery. More practice on this KC is needed for students to reach mastery. Other under-practiced KCs include pentagon-area and triangle-area.

Having students practice less than needed is clearly undesirable in the curriculum. Is over practice necessary? The old idiom "practice makes perfect" suggests that the more practice we do on a skill, the better we can apply the skill. Many teachers believe that giving students more practice problems is beneficial and "would like to have the students work on more practice problems", even when "[students] were not making any mistakes and were progressing through the tutor quickly"[7].

We believe that if the teachers want more problems for their students to practice unmastered KCs or useful KCs not covered by the curriculum, more practice is necessary. To support KC long-term retention, more practice is necessary but needs to be spread on an optimal schedule [1, 14]. In the rectangle-area example, where all the practice for this KC is allocated in a short period, more practice becomes over practice, which is unnecessary after the KC is mastered.

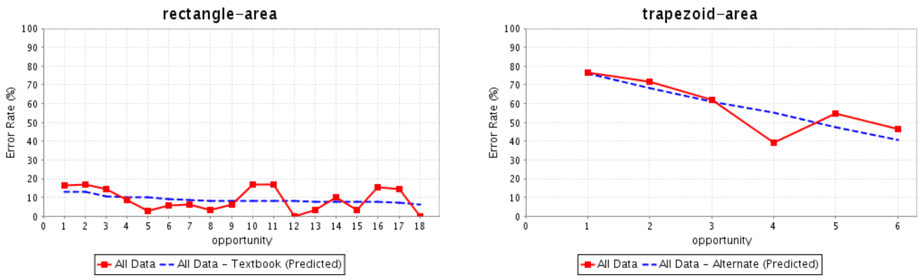


Figure 1 Learning Curve of Rectangle-Area and Trapezoid-Area – The solid lines are the actual error rates over the ordered number of practices. The dotted lines are the error rates predicted by LFA.

1. Design of the Optimized Tutor

What caused the over practice in the Cognitive Tutor curriculum? Cognitive Tutor uses the Knowledge Tracing algorithm to update its estimates of students' mastery of KCs [8]. Based on these estimates, the Tutor chooses to give students the problems with the skills students need to practice more. Table 1 explains the meaning of the four parameters $P(L_0)$, P (T), P(Guess), P(Slip) used in the update. We discovered that the 1997 Tutor used the same set of parameter estimates for all the KCs, as shown in Table 1 column 3. Then we fit the four parameters for each KC with the student log data and used the fit parameters to estimate the amount of practice for these KCs in the same dataset. We found that 58% out of 4102 practices and 31% of 636 exercise questions were done after the students had reached mastery. On the other hand, 119 more practices were needed for all students to master those under-practiced KCs. Although applying the fit parameter estimates on the training data may incur over fitting, the finding does suggest that using a set of carefully calibrated parameter estimates may improve learning efficiency.

To test the effect of calibrated Knowledge Tracing parameters, we planned a study in 2006, when the Geometry Cognitive Tutor had evolved into its 2006 version. The 2006 Tutor breaks the single 1997 area unit into 6 area units (Squares & Rectangles, Parallelograms, Triangles, Trapezoids, Polygons, and Circles), and has a different cognitive model, curriculum design, interface, and student population from its predecessor.

Table 1 Knowledge tracing parameters used in the 1997 Cognitive Geometry Tutor

Parameter	**Meaning (The probability that …)**	**Estimate**
$P(L_0)$	the KC is initially known	0.25
P(T)	the KC transit form an unknown state to a known state	0.2
P(Guess)	a student will apply a KC correctly even if the KC is not learned	0.2
p(Slip)	a student will apply a KC incorrectly even if the KC is learned	0.1

53 KCs are specified in the six units. 32 of them involve calculating perimeters and areas of various shapes. 21 of them involve extracting specific numbers from the text questions and entering them into the tutor interface. The numbers of KCs in each of the 6 units are 19, 7, 7, 8, 4, and 6 respectively. The same set of knowledge tracing parameter estimates shown in Table 1 were used for all its KCs.

Because the 2006 Tutor was just released several weeks before our study, there were no student-log data available from that Tutor to evaluate the KCs empirically. The most recent data available were from the 2005 version of the Cognitive Geometry Tutor, which used the same set of knowledge tracing parameter estimates shown in Table 1. Not surprisingly, we found over-practice and under-practice in that tutor. The over-practiced KCs and the under-practiced KCs are slightly different across the tutors.

Because the cognitive model in the 2005 Tutor is slightly different from the model in the 2006 Tutor, we could not exactly copy those parameter estimates into the 2006 Tutor. To approximate the 2006 estimates, we grouped the KCs in the 2006 Tutor into several homogeneous groups according to their degrees of over-practice in 2005. This is a qualitative mapping of the parameters of one tutor version with a different student population to the parameters of another tutor version. Within each group, KCs share the same parameter estimates. Because we had no relevant information on slips or guesses, we mainly focused on adjusting $P(L_0)$ and P(T) in our study. Between groups, KCs vary mainly on $P(L_0)$. If a KC shows a much higher learning rate P(T) than the original parameter estimate .2, we increased P(T) for that KC. Meanwhile, in order to reduce the danger of under-practice, we reduced all the $P(L_0)$ by a certain amount from the fit estimates. The final parameter estimates in the optimized tutor have the following changes.

- The under-practiced KC (circle-area) is set to a lower $P(L_0) = 0.2$.
- The under-practiced KC with a high learning rate (triangle-area) is set to a lower $P(L_0) = 0.2$, and a higher $P(T) = .5$.
- The slightly-over-practiced KCs (circle-circumference, trapezoid-area, trapezoid-perimeter, triangle-perimeter) are set to $P(L_0) = 0.5$.
- The moderately-over-practiced KCs (parallelogram-area, parallelogram-perimeter, rectangle-area, rectangle-length-or-width, rectangle-perimeter, square-area, square-perimeter, square-side-length) are set to $P(L_0) = 0.7$.
- All the KCs for information extraction are set to $P(L_0) = 0.9$.

Based on these changes, their locations in the units, and the number of KCs in each unit shown, we made the following research hypothesis. Compared with the students in the control condition, students using the optimized tutor will learn the same amount but spend

- much less time in unit 1 and 2 (Squares and Parallelograms);
- moderately less time in unit 3 (Triangles);
- the same amount of time in unit 4 and 5 (Trapezoids and Polygons)
- more time in unit 6 (Circles)

2. Design of the Study

The experiment was conducted in the regular instruction of the geometry course. 110 students from a total of 6 classes in a high school near Pittsburgh participated in the study. They were all taught by the same teacher. The students were randomly assigned to the optimized condition, where they would be using the optimized tutor, or to a control condition where they would be using the original 2006 Tutor with no modifications. We designed a pretest, a post test and a retention test to assess students' ability to solve geometry area and perimeter problems. The test forms were counter-balanced. Each test has 13 – 14 test items including both regular and transfer items.

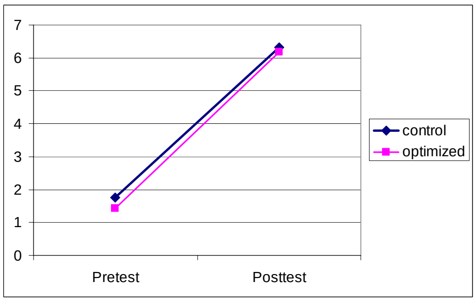
In both conditions, the students took the pretest shortly before they started working on the first unit. Then students spent about 3 days per week on regular classroom instruction and 2 days per week using the Tutor in a computer lab. In the lab, students worked on either the optimized tutor or the original Tutor, according to the condition they had been assigned to. Shortly after finishing the 6th unit, they took the post test. In two weeks after each student finished the post test, we gave each student a retention test. Students' interaction and learning time with the tutor were logged. Due to student attrition and recording errors, 94 students took the pretest; 84 students took the post test; 64 students took the retention test; 73 students took both the pretest and the post test and had valid test scores; and 62 students had valid log data.

3. Results

We found that the optimized group learned as much as the control group but in less time. As seen in Figure 2 (left), the two groups have similar scores in both the pre test and the post test. The amount of learning gain in both groups is approximately 5 points. To further examine the treatment effect, we ran an ANCOVA on the post test scores, with condition as a between subject factor, the pretest scores as a covariate, and an interaction term between the pretest scores and the condition. The post test scores are significantly higher than the pretest, $p < .01$, suggesting that the curriculum overall is effective. Meanwhile neither the condition nor the interaction are significant, $p = 0.772$, and $p = 0.56$ respectively. As shown in the Figure 2 (right), we found no significant difference in the retention test scores ($p = 0.602$, two tailed). The results from the post test and the retention tests suggest that there is no significant difference between the two groups on either of the two tests. Thus, over practice does not lead to a significantly higher learning gain.

The actual learning time in each unit matches our hypotheses. As shown in Table 2, the students in the optimized condition spent less time than the students in the control condition in all the units except in the circle unit. The optimized group saved the most amount of time, 14 minutes, in unit 1 with marginal significance $p = .19$; 5 minutes in unit 2, $p = .01$, and 1.92, 0.49, 0.28 minutes in unit 3, 4, and 5 respectively.

In unit 6, where we lowered $P(L_0)$, the optimized group spent 0.3 more minutes. Notice the percentage of the time saved in each unit. The students saved 30% of tutoring time in unit 2 Parallelogram, and 14% in unit 1 Square. In total students in the optimized condition saved around 22 minutes, an 12% reduction in the total tutoring time.



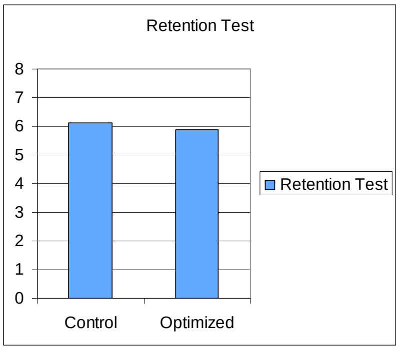


Figure 2 Pretest and post test scores over the two conditions (left) and the retention test scores (right)

Table 2 Time cost in the six tutor curriculum units. The time is in minutes.

	Optimized	**Control**	**Time saved**	**% time saved**	**t Stat**	**P(T<=t) one-tail**
Square	87.16	101.18	14.02	14%	-0.89	0.19
Parallelogram	11.83	16.95	5.12	30%	-2.58	0.01
Triangle	13.03	14.95	1.92	13%	-0.91	0.18
Trapezoid	26.39	26.88	0.49	2%	-0.15	0.44
Polygon	10.58	10.86	0.28	3%	-0.18	0.43
Circle	13.42	13.12	-0.30	-2%	0.18	0.43
Total	162.41	183.93	21.52	12%		

4. Conclusion and Generalizing this Method for Other Tutors

In this study, we presented a method using educational data mining to identify the inefficiency in an ITS. Rather than concluding the research just with the data mining findings, we used the findings to optimize the tutor and showed that students using the optimized tutor saved a significant amount of time compared with the time spent by the control group while both groups achieved a similar learning outcome. While prior studies have demonstrated benefits of cognitive mastery and knowledge tracing, this study is the first we know of that shows using tuned parameters in the tutor yields better learning than hand-set parameters. The knowledge tracing parameters in the existing Cognitive Tutor are hand set.

Although the data was from an earlier version of the tutor with a different student population from that in this study, the fact that efficiency gains were found suggest that this method has potential for generalizing across student population and tutor versions. In addition, this method works for any ITS where student outcomes are labeled with knowledge components and the opportunity to use those KCs. For instance, this method could apply in any other model-tracing tutor or constraint-based tutors [13], where the constraints are the KCs.

5. Other Inefficiencies Found and Future Work

We also found other interesting learning inefficiencies through educational data mining. One tutoring question in the first unit Square involves 4 KCs – Find rectangle length or width from given area, Find rectangle length or width from given perimeter, Find rectangle area, and Find rectangle perimeter. In Cognitive Tutor, each question is presented as an indivisible piece. What if a student has mastered KC 1, 2, 3 but has not mastered KC 4? Unfortunately, the student still has to answer the whole question and practice all four KCs. One way to tackle this inefficiency is to author more tutoring questions, each targeting one specific KC. The tutor would select the corresponding question for an unmastered KC. In the example above, we would create four questions with one for each KC and the tutor would select a question requiring only KC 4. Another solution would be to have the Tutor automatically solve the steps for the mastered KCs and only ask the student to solve the steps for unmastered KCs.

The third and the forth inefficiency relates to cognitive modeling. In the Learning Factors Analysis study [6], we showed that some KCs are better merged into one KC. For example, if we merge "finding the circumference of a circle given diameter (circle-circumference)" and "finding the diameter of a circle given circumference (circle-diameter)" into a single KC circle-CD, the cognitive model fits the log data better and has less complexity. This suggests that as students become more familiar with some low level KCs, the instruction may need to target their aggregated counterparts. Sharp-eye readers may have noticed that unit 1 Square and Rectangle took over 100 minutes in the control condition, greater than the total time of the remaining five units. One cause of this large time consumption is that unit 1 has 19 KCs, over one-third of the total number of KCs in the six units. Due to the similarity between squares and rectangles, we hypothesized that students may only need one representation for both shapes. For example, "square-area" and "rectangle area" may be well represented with one skill. If the hypothesis is true, we can reduce the number of skills in this unit by half. We plan to use LFA to determine appropriateness of merging these skills in a future study.

By extending LFA, Rafferty and Yudelson [15] found that students have different cognitive models -- low-achieving students have more elaborate cognitive models while high-achieving students have more compact models. High-achieving students may do well with even less practice if knowledge tracing uses a compact model for them. To tackle this inefficiency, we need to design a set of representative cognitive models for different groups of students. High-achieving students use a compact model so that they can go through the curriculum more quickly while low-achieving students may be assigned an elaborate model so that they receive more detailed practice. The ambitious solutions to the last two inefficiencies reflect an ITS design principle "adjust the grain size of instruction with learning" [9].

The methods we have discussed have various strengths and implementation difficulties. The method we presented in this paper -- using a set of carefully calibrated knowledge tracing parameter estimates -- can improve learning efficiency without incurring any major tutor modification. It can be easily scaled to other tutor curricula.

The findings support the use of this data mining technique to improve learning efficiency with ITS curricula.

Acknowledgements

This research is sponsored by a National Science Foundation grant SBE-0354420 to the Pittsburgh Science of Learning Center. We thank Carnegie Learning for generously providing the Cognitive Geometry Tutor for our study. We thank Steve Ritter, Joseph Beck, and Albert Colbert for their comments on earlier versions of this paper. We thank Jonathan Steinhart, Christy McGuire, and Frank Baker for their support in tutor modifications and data collection. We thank Alida Skogsholm and Ben Billings for their work on Datashop (http://www.learnlab.org/technologies/datashop), a central repository for educational research data and a set of rich analysis and reporting tools. We thank Colleen Conko for supporting us to use her classes as the study environment. We thank Megan Voorhis for her writing support.

References

[1] J. R. Anderson, J. M. Fincham and S. A. Douglass, *Practice and retention: A unifying analysis* Journal of Experimental Psychology: Learning, Memory, and Cognition, 25 (1999), pp. 1120-1136.

[2] T. Barnes, *The Q-matrix Method: Mining Student Response Data for Knowledge*, *American Association for Artificial Intelligence 2005 Educational Data Mining Workshop*, 2005.

[3] J. E. Beck and B. P. Woolf, *Evaluating tutorial actions with a user model*, *User Modeling and User Adapted Interaction*, 2007.

[4] B. S. Bloom, *The 2 Sigma Problem: The Search for Methods of Group Instruction as Effective as One-to-One Tutoring*, Educational Researcher, 13 (1984), pp. 4-16

[5] H. Cen, K. Koedinger and B. Junker, *Automating cognitive model improvement by a* search and logistic regression*, *American Association for Artificial Intelligence 2005 Workshop on Educational Datamining*, 2005.

[6] H. Cen, K. Koedinger and B. Junker, Learning Factors Analysis - A General Method for Cognitive Model Evaluation and Improvement, 8th International Conference on Intelligent Tutoring Systems, 2006.

[7] J. Cital, *A business email report to Carnegie Learning Inc*, 2006, pp. Email.

[8] A. T. Corbett and J. R. Anderson, *Knowledge tracing: Modeling the acquisition of procedural knowledge*, *User Modeling and User-Adapted Interaction*, 1995, pp. 253-278.

[9] A. T. Corbett, K. Koedinger and J. R. Anderson, *Intelligent Tutorins Systems*, in M. G. Helander, T. K. Landauer and P. Prabhu, eds., *Handbook of Human-Computer Interaction*, Elsevier Science, Amsterdam, The Netherlands, 1997.

[10] K. Draney, P. Pirolli and M. Wilson, *A Measurement Model for a Complex Cognitive Skill*, *Cognitively Diagnostic Assessment*, Erlbaum, Hillsdale, NJ 1995.

[11] K. R. Koedinger and A. T. Corbett, *Cognitive Tutors: Technology Bringing Learning Science to the Classroom*, in K. Sawyer, ed., *The Cambridge Handbook of the Learning Sciences*, Cambridge University Press., 2006, pp. 61-78.

[12] A. Lesgold, S. Lajoie, M. Bunzo and G. Eggan, *Sherlock: A Coached Practice Environment for an Electronics Troubleshooting Job*, in J. H. Larkin and R. W. Chabay, eds., *Computer-assisted instruction and intelligent tutoring systems: shared goals and complementary approaches*, Lawrence Erlbaum Associates, 1988.

[13] A. Mitrovic, M. Mayo, P. Suraweera and B. Martin, *Constraint-based tutors: a success story*, *14th Int. Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems IEA/AIE-2001*, Springer-Verlag Berlin Heidelberg, Budapest, 2001.

[14] P. I. Pavlik and J. R. Anderson, *Practice and Forgetting Effects on Vocabulary Memory: An Activation Based Model of the Spacing Effect* Cognitive Science, 29 (2005), pp. 559-586.

[15] A. Rafferty and M. Yudelson, *Applying LFA for Building Stereotypical Student Models*, *PSLC Summer School 2006 Projects*, Pittsbugh Science of Learning Center, Pittsburgh PA, 2006.